KEY_Lesson16_Pandas-Subsetting-I

December 10, 2021

1 Subsetting Pandas DataFrames I

You now know how to read external datasets into `pandas`. Let's put those skills to use and read in the `tips` dataset again:

[4]:
```
# import the pandas package
import pandas as pd
# set the path
path = 'https://raw.githubusercontent.com/GWC-DCMB/curriculum-notebooks/master/'
# load tips
tips = pd.read_csv(path + 'SampleData/tips.csv')
```

Take a look again at the beginning of the `tips` `DataFrame`:

[5]:
```
# view the beginning of tips
tips.head()
```

[5]:
```
   total_bill   tip     sex smoker  day    time  size
0       16.99  1.01  Female     No  Sun  Dinner     2
1       10.34  1.66    Male     No  Sun  Dinner     3
2       21.01  3.50    Male     No  Sun  Dinner     3
3       23.68  3.31    Male     No  Sun  Dinner     2
4       24.59  3.61  Female     No  Sun  Dinner     4
```

What if we decided we didn't want to keep all of the data recorded in this dataset? To do that, we need to learn how to `subset` `DataFrames`. Subsetting means taking a dataset and pulling out a small portion of it that we're interested in.

First, we'll look at a single column (you can use `head` to keep the printed result short):

[6]:
```
# subset one column
tips['day'].head(10)
```

[6]:
```
0    Sun
1    Sun
2    Sun
3    Sun
4    Sun
5    Sun
```

```
6    Sun
7    Sun
8    Sun
9    Sun
Name: day, dtype: object
```

We use the square brackets [ ] after the name of the DataFrame to tell pandas that we want to look at one of the columns. We put the name of the column in quotes to tell pandas exactly which column we want to look at. Try subsetting the total_bill column:

[7]:
```
# subset the total_bill column
tips['total_bill'].head(10)
```

[7]:
```
0    16.99
1    10.34
2    21.01
3    23.68
4    24.59
5    25.29
6     8.77
7    26.88
8    15.04
9    14.78
Name: total_bill, dtype: float64
```

pandas simply showed us the result of subsetting the column, but it didn't save the result anywhere. Try saving the total_bill column to a new variable, bills:

[8]:
```
# save the total_bill column to a variable
bills = tips['total_bill']
```

We can also pull out multiple columns at a time to create a new DataFrame. If we were only interested in the total_bill and tip, we can subset them like this:

[9]:
```
# subset the columns total_bill and tip
tips[['total_bill', 'tip']].head(10)
```

[9]:
```
   total_bill   tip
0       16.99  1.01
1       10.34  1.66
2       21.01  3.50
3       23.68  3.31
4       24.59  3.61
5       25.29  4.71
6        8.77  2.00
7       26.88  3.12
8       15.04  1.96
9       14.78  3.23
```

Does that look familiar? Instead of putting a single string between the square brackets, we put a whole list of strings – you can tell it's a list by the second set of square brackets. You can also create the list of columns you're interested in and subset the dataframe in two separate steps. This code works exactly the same as what we just did above.

```
[10]: columns = ['total_bill', 'tip']
      tips[columns].head(10)
```

```
[10]:    total_bill   tip
      0       16.99  1.01
      1       10.34  1.66
      2       21.01  3.50
      3       23.68  3.31
      4       24.59  3.61
      5       25.29  4.71
      6        8.77  2.00
      7       26.88  3.12
      8       15.04  1.96
      9       14.78  3.23
```

Now you try: subset the columns `total_bill`, `tip`, and `time` and save the result to a variable called `tips_subset`:

```
[12]: # subset three columns and save to a new variable
      tips_subset = tips[['total_bill', 'tip', 'time']]

      # take a look at the beginning of the new DataFrame
      tips_subset.head()
```

```
[12]:    total_bill   tip    time
      0       16.99  1.01  Dinner
      1       10.34  1.66  Dinner
      2       21.01  3.50  Dinner
      3       23.68  3.31  Dinner
      4       24.59  3.61  Dinner
```

Now we've learned how to subset columns. How do we subset rows? We use a `method` of `DataFrame` called `iloc`. When you see `iloc`, think "index location" – because we want to get the location where the row is a certain index. Let's try it:

```
[13]: # subset a row
      tips.iloc[1]
```

```
[13]: total_bill     10.34
      tip             1.66
      sex             Male
      smoker            No
      day              Sun
```

```
time          Dinner
size               3
Name: 1, dtype: object
```

That showed us the row with an index of 1. Similarly to subsetting columns, we can also subset multiple rows:

[14]:
```
# subset the first three rows
tips.iloc[[0,1,2]]
```

[14]:
```
   total_bill   tip     sex smoker  day    time  size
0       16.99  1.01  Female     No  Sun  Dinner     2
1       10.34  1.66    Male     No  Sun  Dinner     3
2       21.01  3.50    Male     No  Sun  Dinner     3
```

That gave us a smaller `DataFrame` where the rows have an index of 0, 1, or 2. We can do the same thing with slicing syntax:

[15]:
```
# subset multiple rows
tips.iloc[0:3]
```

[15]:
```
   total_bill   tip     sex smoker  day    time  size
0       16.99  1.01  Female     No  Sun  Dinner     2
1       10.34  1.66    Male     No  Sun  Dinner     3
2       21.01  3.50    Male     No  Sun  Dinner     3
```

Notice that this does the same thing as calling `head` with a value of 3:

[16]:
```
# use head
tips.head(3)
```

[16]:
```
   total_bill   tip     sex smoker  day    time  size
0       16.99  1.01  Female     No  Sun  Dinner     2
1       10.34  1.66    Male     No  Sun  Dinner     3
2       21.01  3.50    Male     No  Sun  Dinner     3
```

What if we want to grab some rows in the middle of the `DataFrame`? Try subsetting the 100th through 105th row. Hint: Don't forget that counting starts at `0` in Python!

[17]:
```
# subset the 100th row through the 105th row
tips.iloc[99:105]
```

[17]:
```
     total_bill   tip     sex smoker  day    time  size
99       12.46  1.50  Male     No  Fri  Dinner     2
100        11.35  2.50  Female     Yes  Fri  Dinner     2
101        15.38  3.00  Female     Yes  Fri  Dinner     2
102        44.30  2.50  Female     Yes  Sat  Dinner     3
103        22.42  3.48  Female     Yes  Sat  Dinner     2
```

```
104        20.92  4.08  Female     No  Sat  Dinner     2
```

Congrats on making it to the end of this lesson – we learned a lot!

- How to use square brackets `[]` to subset columns.
- How to use `iloc` to subset rows.